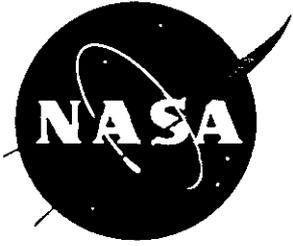




Preliminary Control System Design and Analysis for the Space Station Furnace Facility Thermal Control System

M.E. Jackson



Preliminary Control System Design and Analysis for the Space Station Furnace Facility Thermal Control System

M.E. Jackson
Marshall Space Flight Center • MSFC, Alabama

TABLE OF CONTENTS

	Page
INTRODUCTION	1
PURPOSE AND SCOPE OF ANALYSIS	1
CONTROL SYSTEM MATHEMATICAL MODEL	2
Flow Path Model	2
Source Model	2
CONTROL SYSTEM DESIGN	3
Pressure Controller	4
Flow Controller	4
CONTROL SYSTEM ANALYSIS	4
CONTROL SYSTEM SIMULATIONS	5
CONCLUSIONS.....	5
REFERENCES.....	13
APPENDIX	15

LIST OF ILLUSTRATIONS

Figure	Title	Page
1.	TCS design layout.....	6
2.	TCS fluid circuit.....	7
3.	Pump pressure map	7
4.	Pressure controller block diagrams	8
5.	Flow controller block diagrams	8
6.	Pressure frequency response	9
7.	Flow frequency response	9
8.	EM1A flow and valve angle plots.....	10
9.	Core flow and valve angle plots.....	10
10.	EM2B flow and valve angle plots.....	11
11.	Pressure drop and pump speed plots	11
12.	Total system flow plot.....	12
13.	Static subsystem path's flow plots	12

TECHNICAL MEMORANDUM

PRELIMINARY CONTROL SYSTEM DESIGN AND ANALYSIS FOR THE SPACE STATION FURNACE FACILITY THERMAL CONTROL SYSTEM

INTRODUCTION

The Space Station Furnace Facility (SSFF) is a facility for materials research in the microgravity environment of the international Space Station Alpha United States Laboratory. The SSFF provides the necessary core systems to operate various materials processing furnaces. The thermal control system (TCS) is defined as one of the core systems, and its function is to collect excess heat from furnaces and to provide precise cold temperature control of components and of certain furnace zones. The TCS utilizes single-phase water as its cooling medium and consists of piping, heat exchangers, cold plates, valves, flow sensors, temperature sensors, pressure sensors, and a pump. The TCS design layout is shown in figure 1. Combinations of the subsystems are plumbed in parallel to help heat rejection controllability.

The specific control system objective for the TCS is to (1) control the total pressure drop across the parallel system and (2) track defined flow rate profiles in each parallel subsystem path. Two control inputs are employed to achieve the control system objective. One control input is the pump speed command, which is used to modulate pump speed to maintain the desired system pressure drop. This serves the purpose of decoupling the flow dynamics between the respective parallel paths. The other control input is the valve speed command, which is used to modulate the flow control valve to track defined flow rate profiles in each subsystem path. Tracking the desired flow profiles assures that subsystem temperatures are maintained.

PURPOSE AND SCOPE OF ANALYSIS

The SSFF TCS control system analysis was performed to justify and fine-tune digital control algorithms that generate the control inputs. The analysis is an ongoing activity consisting of model development, model validation, and open- and closed-loop analyses utilizing frequency, time, and discrete domain techniques. This report serves as the documentation of the TCS control system design and analysis performed prior to the SSFF preliminary design review. The mathematical modeling, control system designs, and control system analyses were performed using Marshall System for Aerospace Simulation (MARSYAS), a control system analysis and time simulation package that was developed by the Control Systems Division at Marshall Space Flight Center. The following sections are included in this report: the description of the TCS mathematical model; the control system design, both continuous and discrete; the control system analysis including stability margins; the system performance analysis including time simulations; and the MARSYAS code used in the simulations and analyses.

CONTROL SYSTEM MATHEMATICAL MODEL

The TCS has a pump source that forces cold water through six subsystem flow paths (fig. 1 shows eight paths, but only six are operated simultaneously) plumbed in parallel. The water is used to collect waste energy and to control thermal environments of the various subsystems. The mathematical model of the TCS is divided into two submodels: the flow path model and the source model.

Flow Path Model

Each flow path consists of components (cold plates, heat exchangers, flex lines, quick disconnects, furnace cooling jackets, etc.) that are plumbed in series with a flow control actuator and valve. The flow model has an equation that describes the mass flow rate of water and two equations that describe the flow control actuator and valve. The mass flow rate equation (1) is derived from an equivalent circuit (fig. 2) analogy using lumped parameters for flow resistance, inertance, and capacitance.¹ The flow equation is written as:

$$\frac{dW_i}{dt} = \frac{1}{L_i} \left\{ - \left[(\alpha e^{-\beta\theta_i} + \delta) + k_i \right] W_i^2 + P \right\}, \quad \forall i = 1,6, \quad (1)$$

where W_i is the mass flow rate through each path, θ_i is the angle of the flow control valve in each path, ω_i is the velocity of the flow control valve in each path, L_i is the lumped flow inertance through each path, α , β , and δ are the valve resistance constants, R_i is the lumped flow resistance constant for each path, and P is the system pressure drop across the paths.

The flow control actuator consists of an analog speed controller and a brushless dc motor. The flow control actuator and valve are connected through a shaft and together provide precise control of the flow profiles in each path. The valve equation (2) is derived from the idea that the valve acts as an ideal integrator of its commanded speed. The valve equation is written as:

$$\frac{d\theta_i}{dt} = \omega_i, \quad \forall i = 1,6. \quad (2)$$

The speed controller equation (3) is approximated from the idea that the controller achieves the commanded valve speed with a slight time delay. The speed controller equation is written as:

$$\frac{d\omega_i}{dt} = \frac{1}{\tau} (k_v u_i - \omega_i), \quad \forall i = 1,6, \quad (3)$$

where τ is the time constant delay of the commanded valve speed, k_v is the conversion constant between voltage and valve speed, and u_i is the valve input voltage command for each path.

Source Model

The source model consists of the system flow and pressure drop, pump motor, pump speed controller, and system pressure controller equations. The system flow equation (4) and the system pressure

drop equation (5) are derived using an equivalent circuit analogy.¹ The system flow equation is written as:

$$\frac{dW_T}{dt} = \frac{1}{L_{eq}} \left[-R_{eq} W_T^2 + P_p(W_T, \omega_p) + P \right], \quad (4)$$

where W_T is the system flow through the pump, P is the system pressure drop across the paths, L_{eq} is the equivalent system flow inertance, R_{eq} is the equivalent system flow resistance, ω_p is the speed of the pump, and $P_p(W_T, \omega_p)$ is the pump map function (fig. 3). The system pressure drop equation is written as:

$$\frac{dP}{dt} = \frac{2}{C_{eq}} \left(W_T - \sum_{i=1}^6 W_i \right), \quad (5)$$

where C_{eq} is the equivalent system flow capacitance, and W_i is the flow through each subsystem path.

The dynamic model representation of the pump and the corresponding parameter data are obtained from the pump vendor, Allied Signal.² The pump motor and controller equations are written as follows:

$$\frac{d\omega_p}{dt} = \frac{1}{j_{ml}} (k_t I_a - k_1 \omega_p^2), \quad (6)$$

$$\frac{dI_a}{dt} = \frac{1}{L_m} (-R_m I_a - k_e \omega_p - k_2 (P - P_{set}) - k_3 \omega_{fbk} + E X_c), \quad (7)$$

$$\frac{d\omega_{fbk}}{dt} = \frac{1}{\tau_f} (\omega_p - \omega_{fbk}), \quad (8)$$

$$\frac{dX_c}{dt} = F X_c + G \cdot [\omega_{fbk} \quad P \quad P_{set}]^T, \quad (9)$$

where ω_p is the pump speed, I_a is the pump motor armature current, ω_{fbk} is the pump speed feedback, X_c is the 2×1 row-vector of controller states, j_{ml} is the rotor plus load inertia, k_t is the current to motor torque gain, k_1 is the speed to load torque gain, L_m is the pump motor inductance, R_m is the pump motor resistance, k_e is the motor back emf gain, k_2 is the controller feed forward gain, k_3 is the controller feed forward gain, E is the 1×2 row-vector of controller gains, τ_f is the speed filter time constant, F is the 2×2 matrix of controller gains, and G is the 2×3 plant state input matrix.

CONTROL SYSTEM DESIGN

The specific control system design objective is to (1) control the total pressure drop across the parallel system and (2) track defined flow rate profiles in each parallel subsystem path. Two controllers are designed to achieve this objective. The first is a pressure controller that is used to modulate the pump speed to maintain the total pressure drop. The second is a flow controller that is used to modulate the flow control valve to track defined flow profiles in each subsystem path.

Pressure Controller

The coupling effect between subsystem paths is the most significant disturbance to controlling flow. For instance, flow rate modulation in one subsystem path causes a disturbance to the other subsystem paths. This so-called coupling effect can be seen mathematically from the plant equations. When written in matrix notation, the plant's flow equations have off-diagonal terms in pressure. These terms represent the coupling between subsystem paths. To minimize this effect, the pressure between the subsystem paths is controlled to a constant. The continuous and discrete control system designs to perform this pressure control are shown in figure 4. The continuous system was designed and then converted to a discrete equivalent system³ with a sample rate of 1 s.

The pressure control system design has a lead-lag filter that improves system stability and response through the placement of poles and zeros in the frequency domain. The prime purpose of a lead filter is to add phase near the crossover frequency in order to increase the phase margin. Accompanied with this phase lead is a gain increase that will increase the crossover frequency. The basic idea behind the lag filter design is to reduce the gain at "middle" frequencies in order to reduce the crossover frequency to a lower value than for the uncompensated system. Combining the lead and lag filters provides attenuation below the crossover frequency to decrease the phase lag at crossover and phase lead closer to the crossover frequency in order to increase the phase lead of the uncompensated system at the crossover frequency.⁴

The output of the controller is the pump speed command that is limited and held for a 1-s sample delay. The MARSYAS code implementing the continuous and discrete controllers is listed in the appendix.

Flow Controller

Performance requirements for flow control such as rise times, overshoots, and settling times are not specified for a good reason—they are not critical. Because the dynamic lag between flows and temperatures is so large (minutes), transient variations in flow and pressure do not significantly affect subsystem temperatures. However, long-term variations in flow do affect temperatures. With this in mind, the control goal is to design a very stable control system that is insensitive to disturbances and has good steady-state accuracy.

The continuous and discrete control system designs for flow rate control to each EM and to the core rack are shown in figure 5. The continuous system was designed and then converted to a discrete equivalent system³ with a sample rate of 1 s. The flow set point and the flow error signals are low pass filtered to smooth out the system's response to disturbances. The flow control system design has a proportional-plus-derivative (PD) control algorithm⁵ that achieves the desired flow tracking performance. The valve model has a free integrator, which coupled with the PD controller, achieves zero steady-state flow tracking error. A valve command limiter and a computer processing delay are also included. The MARSYAS code implementing the continuous and discrete controllers is listed in the appendix.

CONTROL SYSTEM ANALYSIS

An analysis is performed to determine the system's sensitivity to variations in pressure and flow. The control system shown in figure 4 is used to analyze the system's sensitivity to pressure variations.

The feedback control loop is "opened at" the Err variable and a new input and output created. The transfer function between the new input and output is generated and a frequency response run. The response is shown in figure 6. From this plot, the gain and phase margins are obtained as 20 dB and 110°, respectively.

The gain margin is defined as the amount of gain the system would require to cross 0 dB at 180° of phase. The phase margin is defined as the amount of phase the system would require to cross 180° at 0 dB of gain. Gain and phase margins quantify the relative stability or sensitivity that the system has to a certain parameter.

The control system shown in figure 5 is used to analyze the system's sensitivity to variations in flow. The feedback control loop is "opened at" the Err variable and a new input and output created. This process is repeated for all three flow control loops (two EM's and core). The resulting frequency response, shown in figure 7, is representative of all three flow control systems (EM1A, EM2B, and the core). The plot shows a gain and phase margin of 50 dB and 78°, respectively. These large stability margins indicate that the system is very stable to variations in flow and pressure.

CONTROL SYSTEM SIMULATIONS

The integrated TCS is simulated on the digital computer. Active control is delivered to the system at a sample rate of 1 s. For this discussion, the pressure and flow nomenclature shown in figure 2 is used. For this simulation, the control objective is to track flow rate profiles for W2, W13, and W18, which represent the flows through EM1A, the core, and EM2B, respectively, and to maintain the static flows for W3, W6, and W17, which represent the flows through the constant flow subsystem paths. To help decouple the flow dynamics between the subsystem paths, the differential pressure between P1 and P10 is controlled to 10 lb/in².

The model is initialized at a steady-state operating point and simulated for 6 min. The input to the model is simultaneous changes in all three flow rate profiles for W2, W13, and W18. The solid and dotted lines on the top graphs in figures 8, 9, and 10 are plots of the flows and profiles, respectively. The bottom graphs in figures 8, 9, and 10 are plots of the respective valve positions that are modulated by the flow control algorithm to track the flow profiles. The top graph in figure 11 is the plot of the controlled system pressure drop. The bottom graph in figure 11 is the commanded pump speed to maintain the pressure drop. The graph in figure 12 is a plot of the total system flow through the pump, and the graphs in figure 13 are the plots of the static flows in the constant flow subsystem paths.

CONCLUSIONS

The nonlinear mathematical model of the TCS has been developed and presented in this report. A very robust control system has been designed to maintain flow profiles and pressures in the TCS in the face of simultaneous disturbances. Operationally, the TCS will not experience the simultaneous high-frequency flow profile changes applied in the simulation; nevertheless, these disturbances were applied to assess its performance in a worst-case scenario. The simulation demonstrates that the design goal of a very stable control system that is insensitive to disturbances and has good steady-state accuracy is achieved. Control system analyses and simulation runs show that the system is very stable and has satisfactory transient and steady-state response. The control system designs and analyses will continue to mature as model validations are performed.

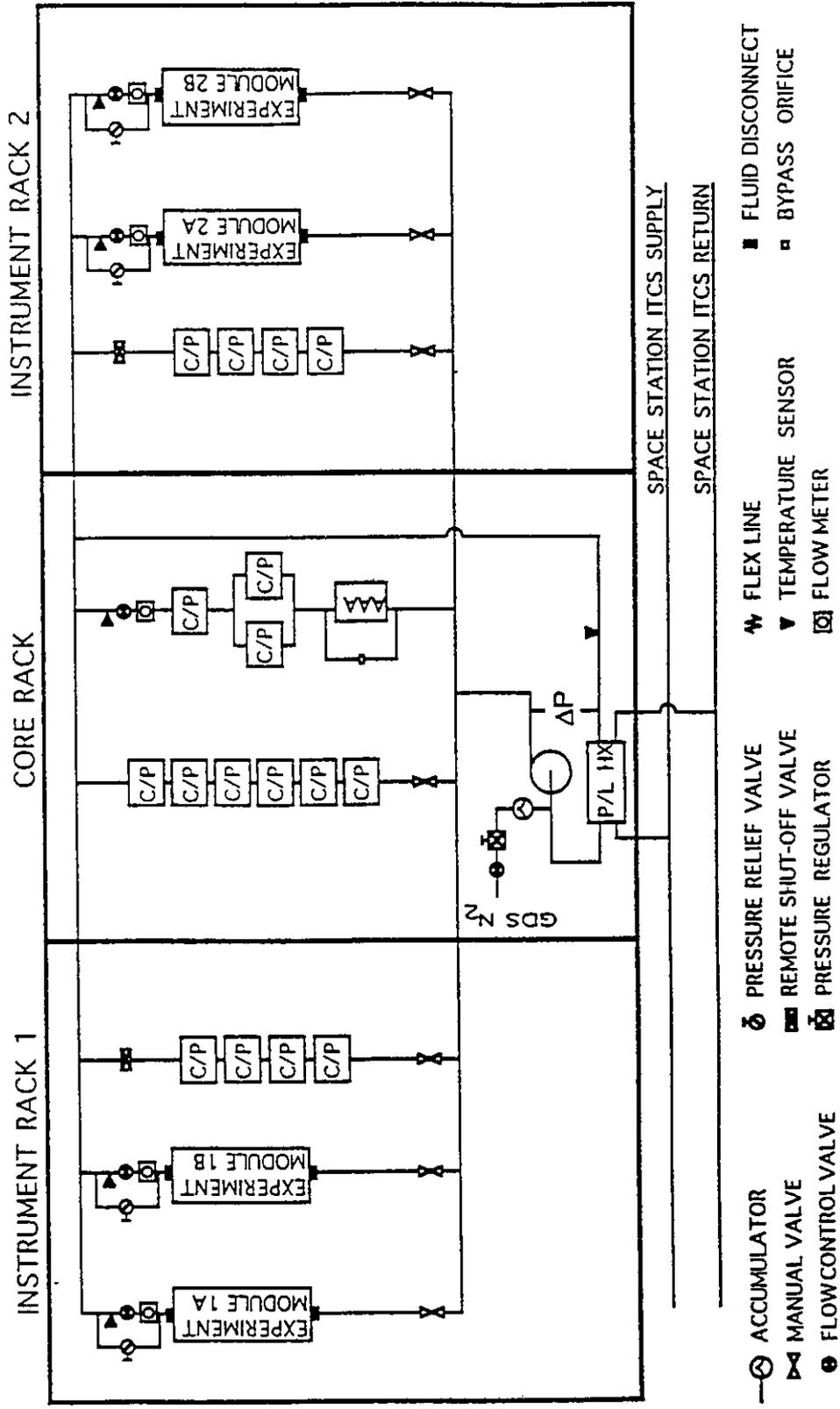


Figure 1. TCS design layout.

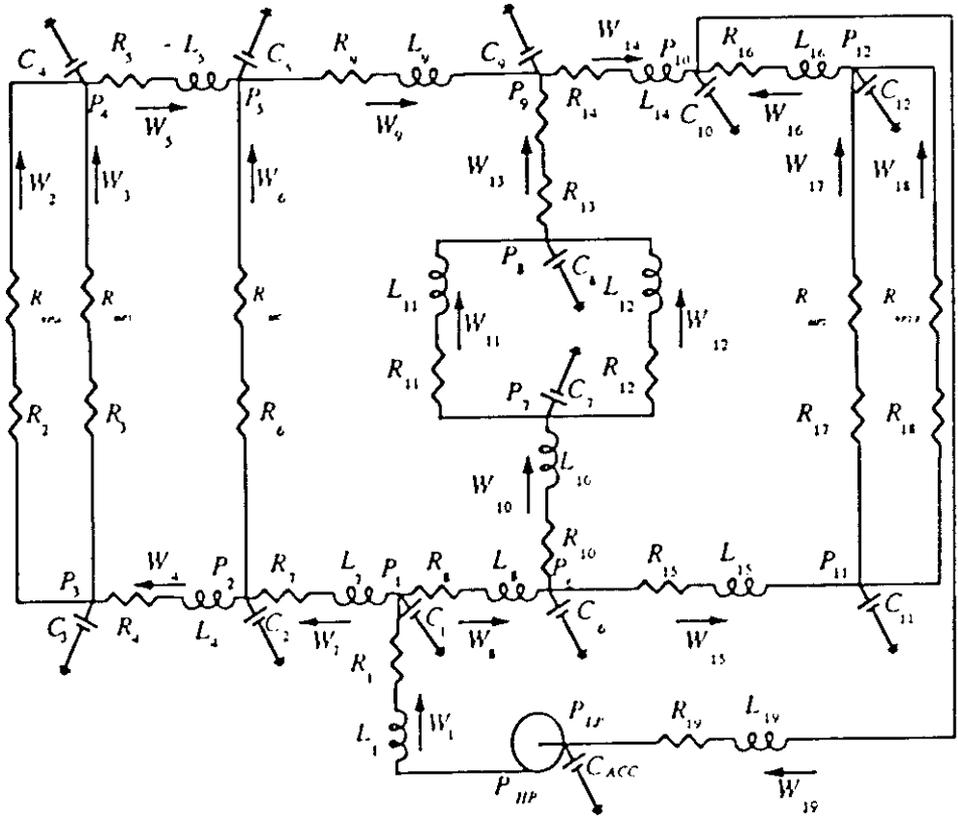


Figure 2. TCS fluid circuit.

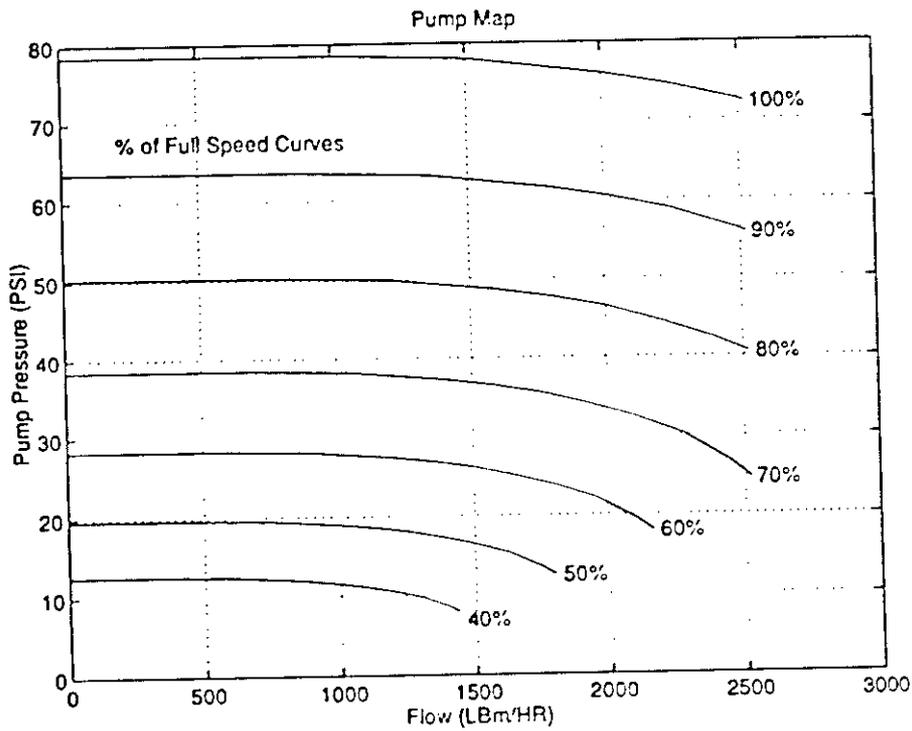
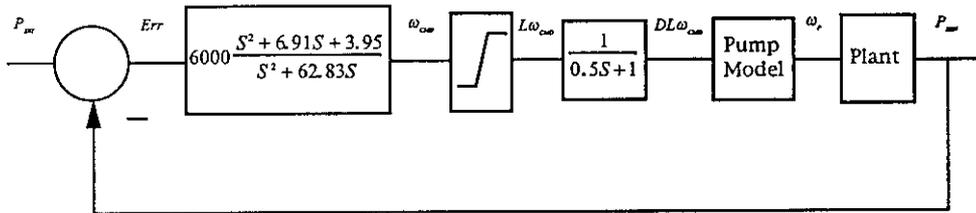


Figure 3. Pump pressure map.

Continuous Pressure Controller



Discrete Pressure Controller

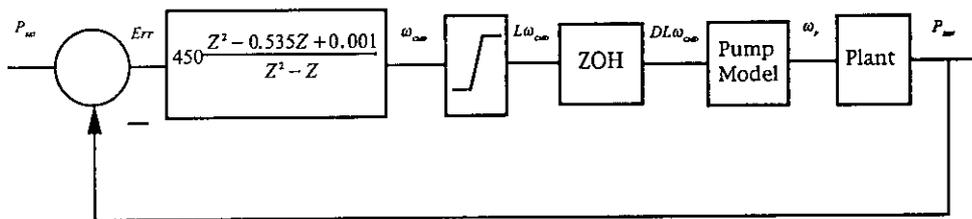
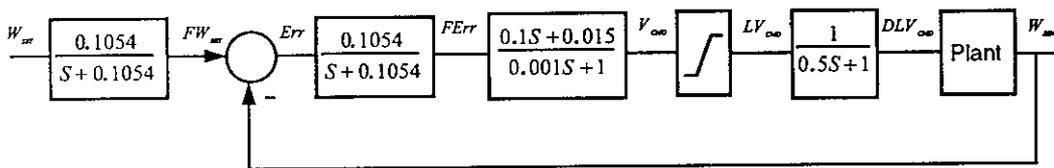


Figure 4. Pressure controller block diagrams.

Continuous Flow Controller



Discrete Flow Controller

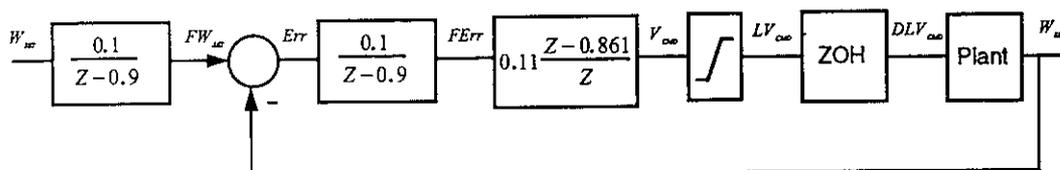


Figure 5. Flow controller block diagrams.

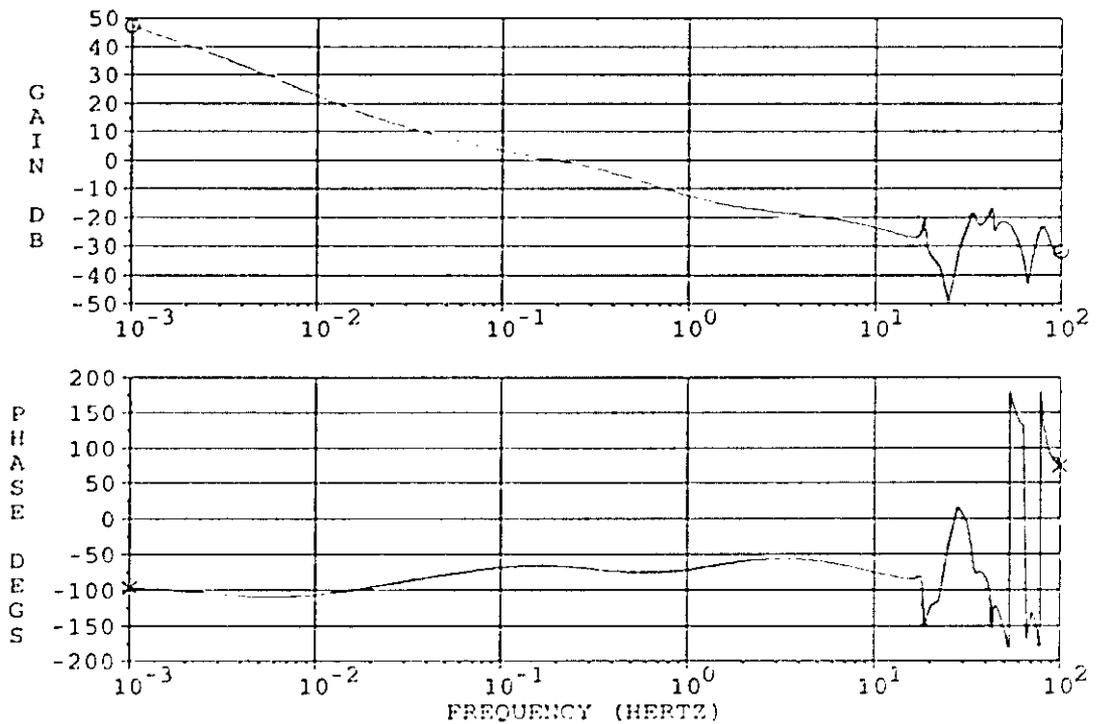


Figure 6. Pressure frequency response.

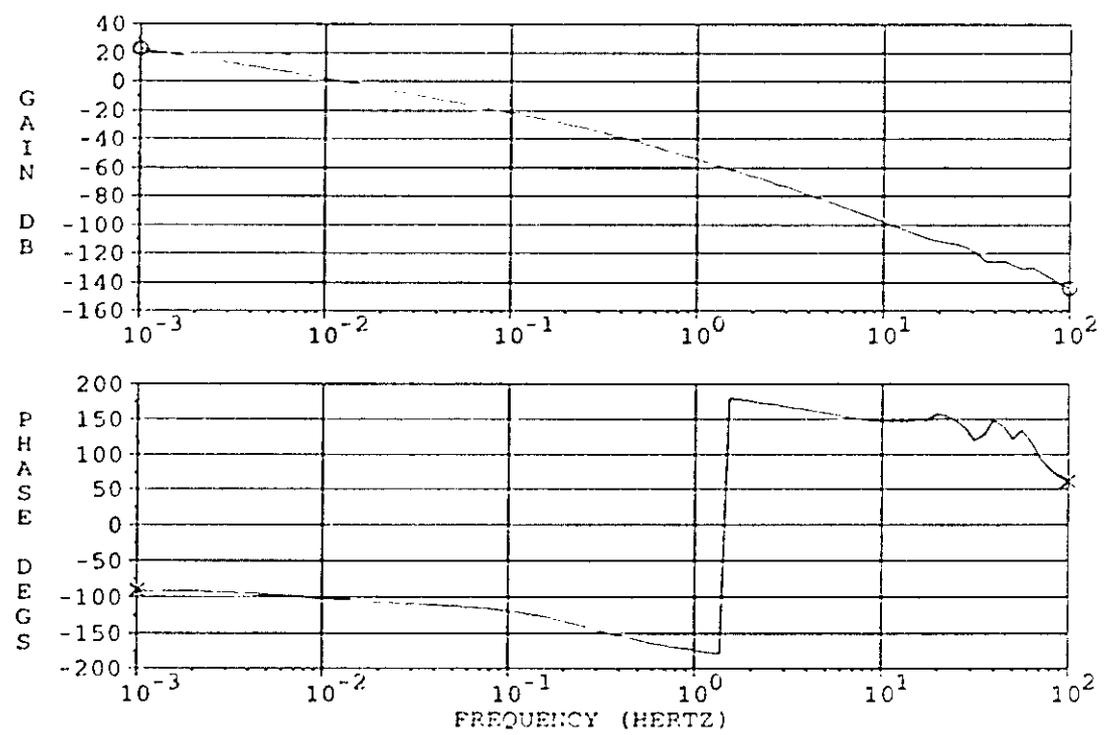


Figure 7. Flow frequency response.

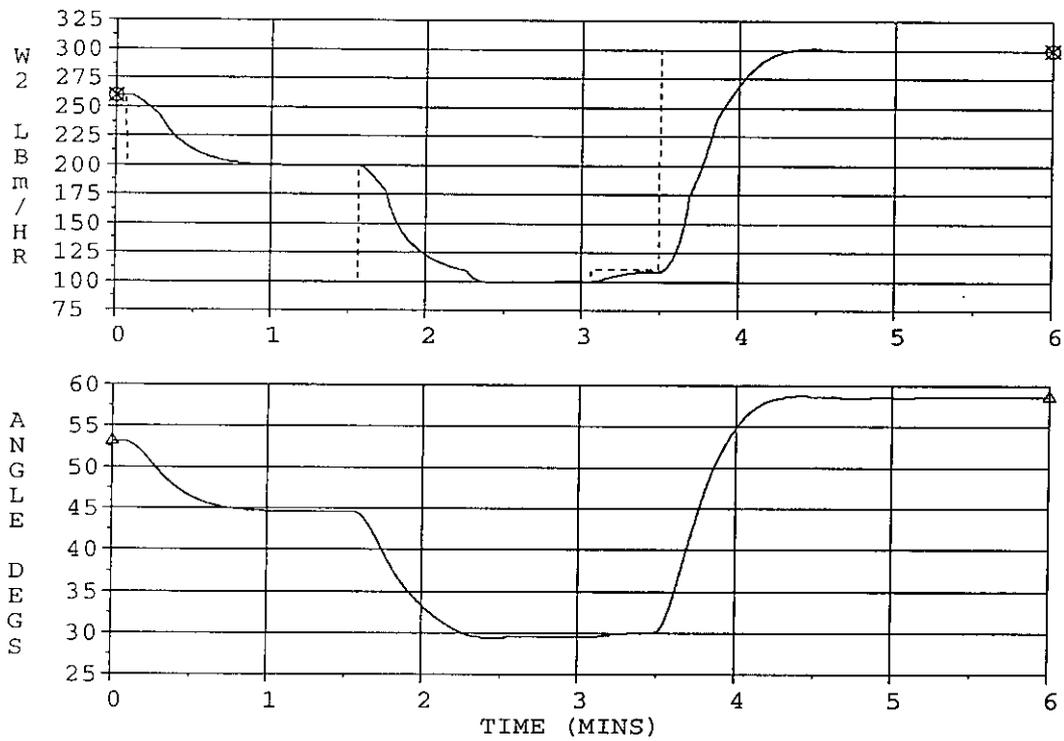


Figure 8. EM1A flow and valve angle plots.

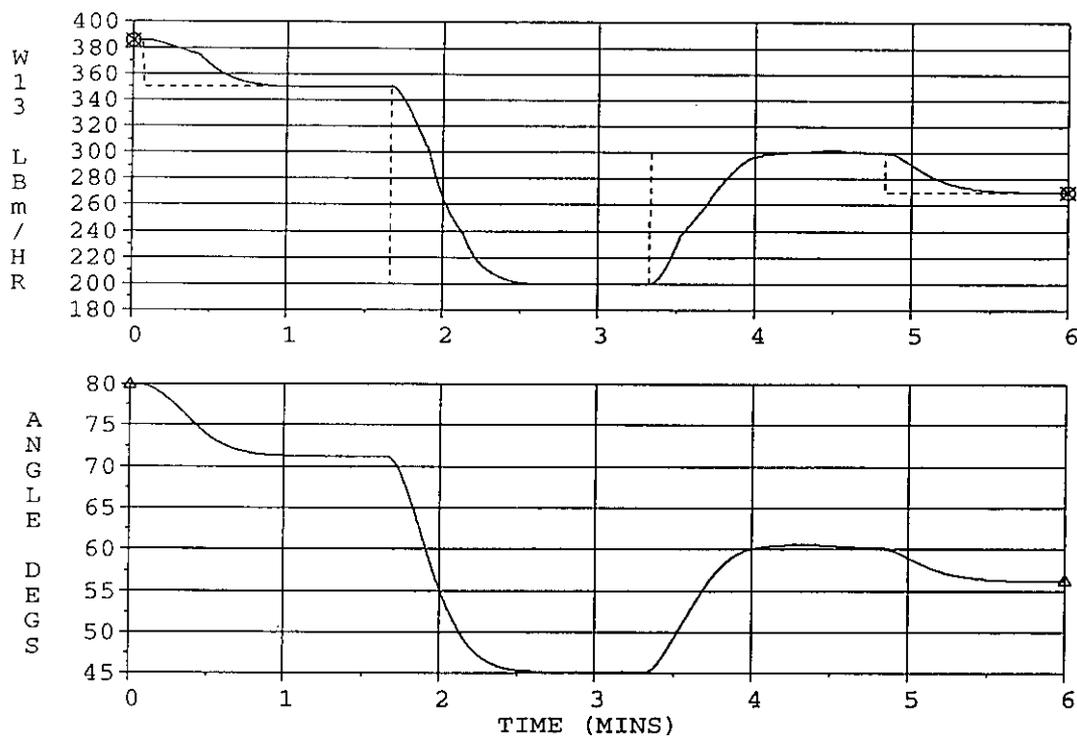


Figure 9. Core flow and valve angle plots.

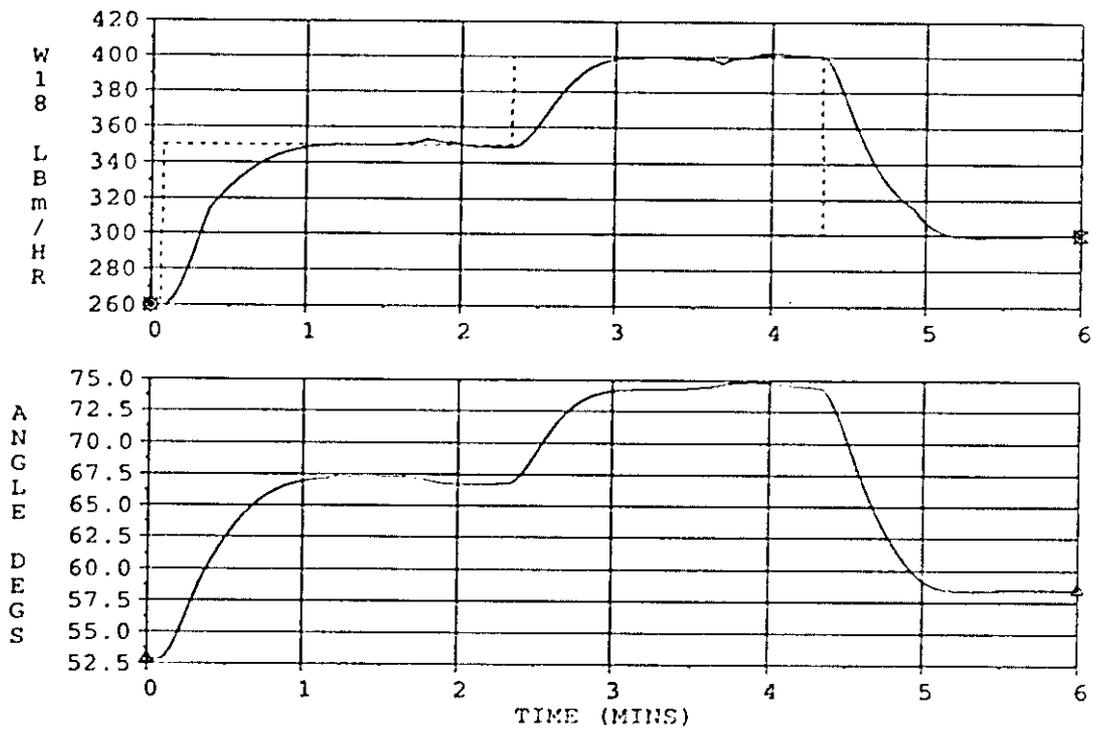


Figure 10. EM2B flow and valve angle plots.

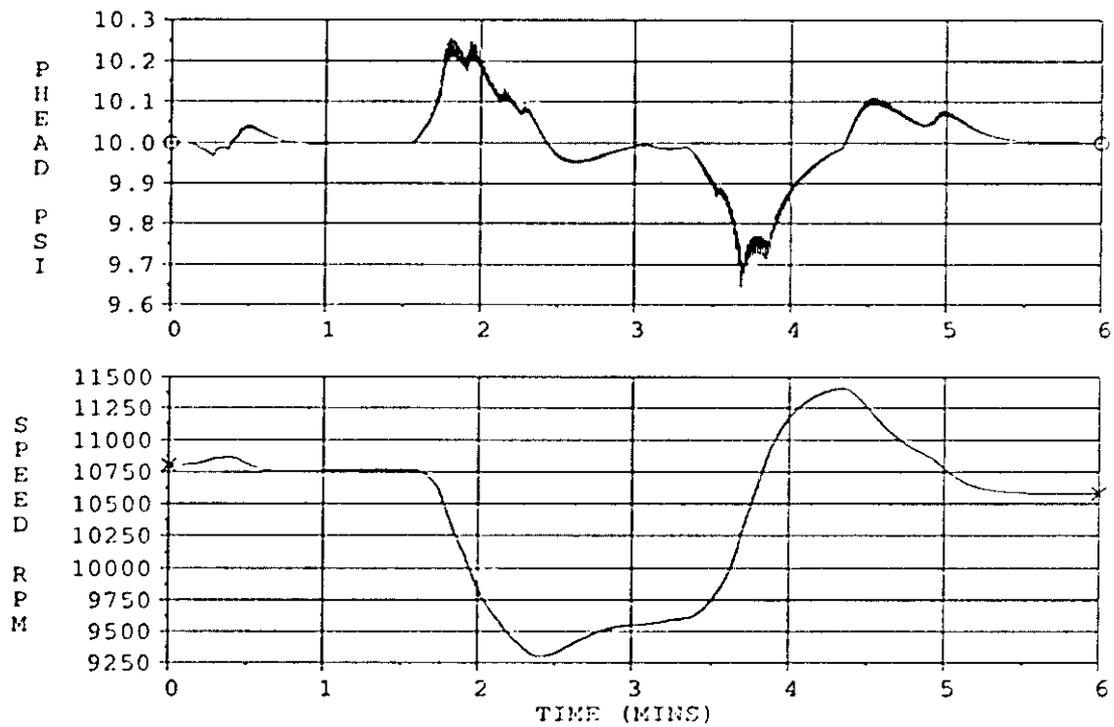


Figure 11. Pressure drop and pump speed plots.

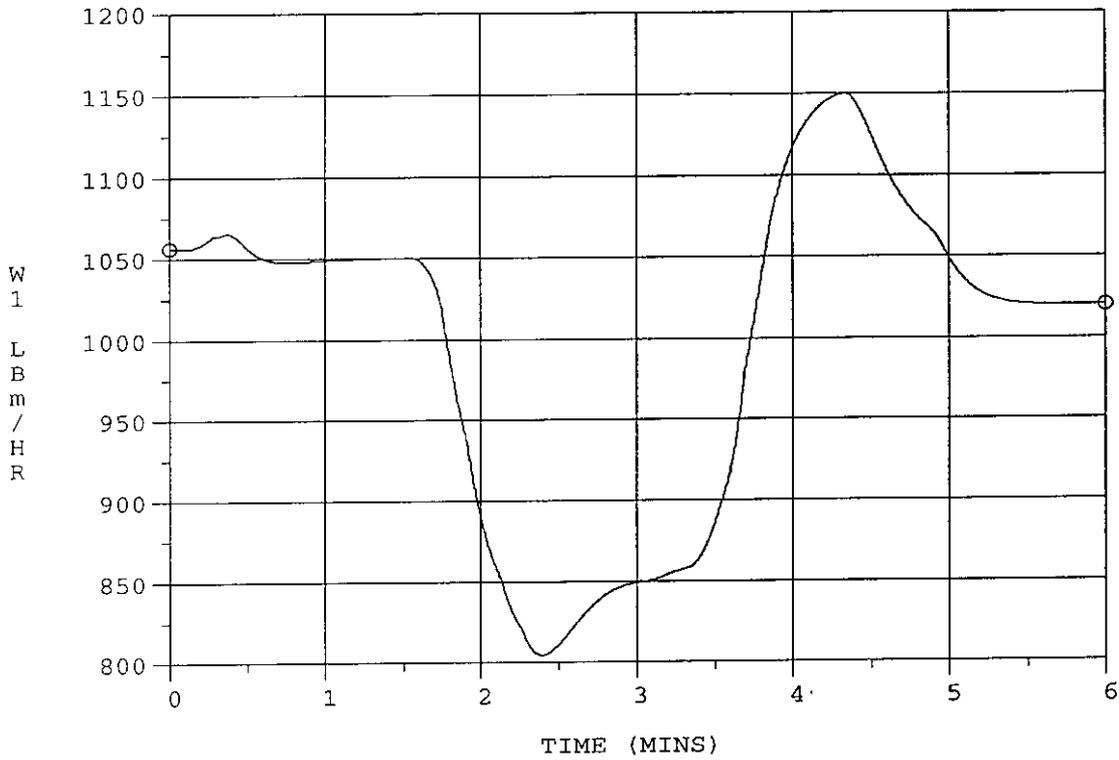


Figure 12. Total system flow plot.

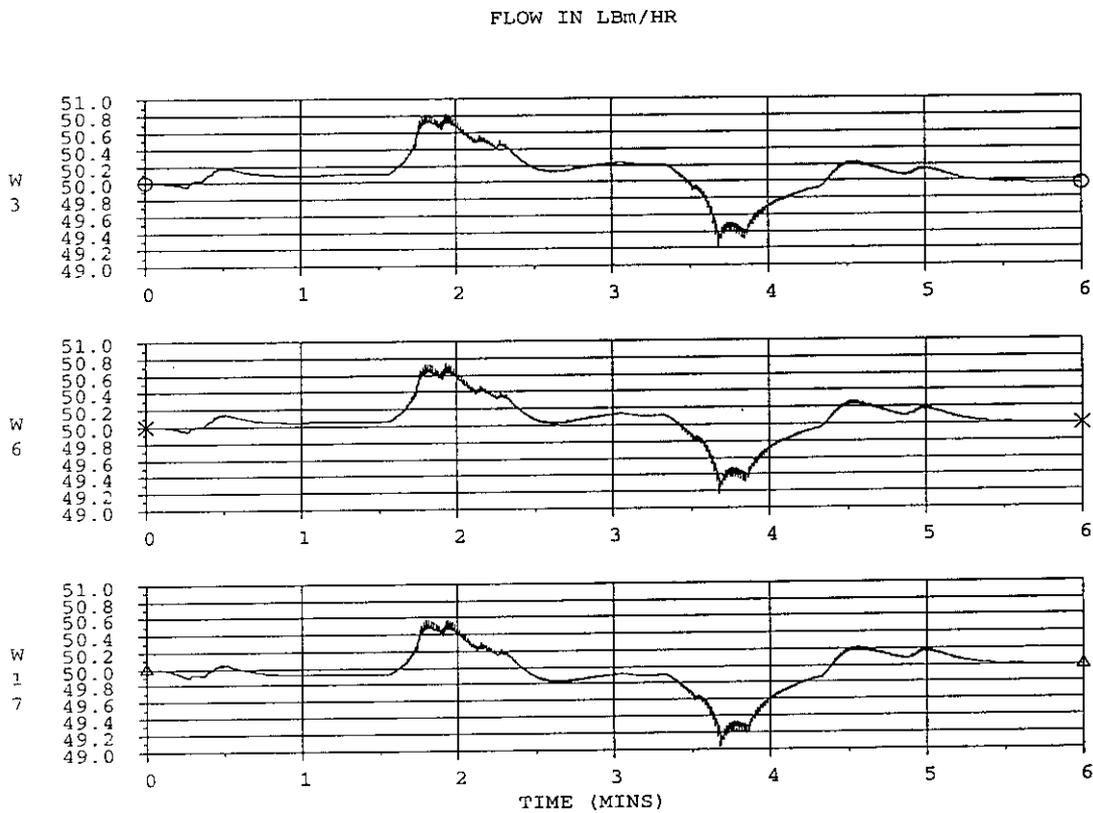


Figure 13. Static subsystem path's flow plots.

REFERENCES

1. Ogata, K.: "System Dynamics." Prentice Hall, Inc., Englewood Cliffs, NJ, 1978.
2. Allied Signal Aerospace Systems and Equipment: "Pump/Fan Motor Controller Dynamic Simulation Model." Memorandum, August 10, 1993.
3. Ogata, K.: "Discrete-Time Control Systems." Prentice Hall, Inc., Englewood Cliffs, NJ, 1987.
4. Sage, A.: "Linear Systems Control." Matrix Publishers, Champaign, IL, 1978.
5. Phillips, C.: "Feedback Control Systems." Prentice Hall, Inc., Englewood Cliffs, NJ, 1988.

APPENDIX

**** main ****

- * This file is located in /home/jacksme/marsyas/furn/main on host ed14j.
- * It contains the main program for the SSFF TCS simulation. It also
- * contains defined constants and the respective inputs and outputs of
- * the main model and the various submodels \$

- * program global constant definitions \$

CONSTANT: PI = 3.1415927 \$

- * density of water (lbm/ft³) \$

: RHO = 62.268 \$

- * conversion constant (lbm-ft / sec²-lbf) \$

: GC = 32.174 \$

- * inner diameter for components (ft) \$

: DC = 0.305/12 \$

- * inner diameter for pipes (ft) \$

: DI = 0.5/12 \$

- * algebraic loop breaker delay time (secs) \$

: TDELAY = 0.005 \$

- * integration timestep (secs) \$

: TSTEP = 0.001 \$

- * simulation stoppage time (secs) \$

: STOPT = 0.0 \$

- * include the file /home/jacksme/marsyas/furn/defines which has all

- * function call definitions \$

INCLUDE: "defines" \$

- * main model definition \$

- * inputs:

- * amf1 - valve angle for static path in ir1 (degs) \$

- * amc - valve angle for static path in core (degs) \$

- * amf2 - valve angle for static path in ir2 (degs) \$

- * pump\ref\speed - 100 percent pump speed (rpm) \$

- * t - water temperature (degs f) \$

- * pheadset - system pressure set point (psi) \$

- * w2set - flow set point through em1a in ir1 (lbm/hr) \$

- * w13set - flow set point through core (lbm/hr) \$

- * w18set - flow set point through em2b in ir2 (lbm/hr) \$

- * output definitions:

- * time\min - simulation time (mins) \$

- * phead - system pressure (psi) \$

- * pump\speed - pump speed (rpm) \$

- * w2set - flow set point through em1a in ir1 (lbm/hr) \$

- * w2 - flow through em1a in ir1 (lbm/hr) \$

- * em1a\valve\angle - valve position in path em1a of ir1 (degs) \$

- * w13set - flow set point through core (lbm/hr) \$

- * w13 - flow through core (lbm/hr) \$

* core\valve\angle - valve position in core (degs) \$
 * w18set - flow set point through em2b in ir2 (lbm/hr) \$
 * w18 - flow through em2b in ir2 (lbm/hr) \$
 * em2b\valve\angle - valve position in path em2b of ir2 (degs) \$
 * w1 - total system flow through pump (lbm/hr) \$
 * w3 - flow through static path in ir1 (lbm/hr) \$
 * w6 - flow through static path in core (lbm/hr) \$
 * w17 - flow through static path in ir2 (lbm/hr) \$
 MODEL: MAIN \$
 INPUTS: AMF1,AMC,AMF2,PUMP\REF\SPEED,I,PHEADSET,W2SET,W13SET,W18SET \$
 OUTPUTS: TIME\MIN,PHEAD,PUMP\SPEED,W2SET,W2,EM1A\VALVE\ANGLE,W13SET,W1,
 CORE\VALVE\ANGLE,W18SET,W18,EM2B\VALVE\ANGLE,W1,W3,W6,W17 \$

* pump speed controller submodel definition \$
 * inputs:
 * pump\speed\cmd - pump speed command from pressure controller (rads/sec) \$
 * pump\speed\sen - pump speed (rads/sec) \$
 * outputs:
 * pump\volt\cmd - pump motor voltage command (volts) \$
 SUBMODEL: PUMP\SPEED\CONTROLLER;
 INPUTS: PUMP\SPEED\CMD,PUMP\SPEED\SEN;
 OUTPUTS: PUMP\VOLT\CMD \$

* pump motor submodel definition \$
 * input:
 * pump\volt\cmd - pump motor voltage command (volts) \$
 * outputs:
 * pump\speed - pump speed (rpm) \$
 * pump\speed\sen - pump speed (rads/sec) \$
 * pump\curr\sen - pump motor armature current (amps) \$
 SUBMODEL: PUMP\MOTOR;
 INPUTS: PUMP\VOLT\CMD;
 OUTPUTS: PUMP\SPEED,PUMP\SPEED\SEN,PUMP\CURR\SEN \$

* flow control valve\actuator submodel definitions \$
 * inputs:
 * 'path\valve\cmd - valve command (volts) \$
 * outputs:
 * 'path\valve\angle - valve angle (degs) \$
 SUBMODEL: EM1A\VALVE\ACTUATOR ;
 INPUTS: EM1A\VALVE\CMD ;
 OUTPUTS: EM1A\VALVE\ANGLE \$

SUBMODEL: CORE\VALVE\ACTUATOR ;
 INPUTS: CORE\VALVE\CMD ;
 OUTPUTS: CORE\VALVE\ANGLE \$

SUBMODEL: EM2B\VALVE\ACTUATOR ;
 INPUTS: EM2B\VALVE\CMD ;
 OUTPUTS: EM2B\VALVE\ANGLE \$

* continuous and discrete pressure controller submodel definitions \$

* inputs:

* phead - system pressure (psi) \$

* pheadset - system pressure setpoint (psi) \$

* outputs:

* pump\speed\cmd - pump speed command (rads/sec) \$

SUBMODEL: PRESSURE\CONTROLLER\CONTINUOUS;

INPUTS: PHEAD,PHEADSET ;

OUTPUTS: PUMP\SPEED\CMD \$

SUBMODEL: PRESSURE\CONTROLLER\DISCRETE;

INPUTS: PHEAD,PHEADSET;

OUTPUTS: PUMP\SPEED\CMD \$

* continuous and discrete flow controller submodel definitions\$

* inputs:

* w# - path flow (lbm/hr) \$

* w#set - path flow set point (lbm/hr) \$

* outputs:

* 'path\valve\cmd - path valve command (volts) \$

SUBMODEL: EM1A\FLOW\CONTROLLER\CONTINUOUS ;

INPUTS: W2,W2SET ;

OUTPUTS: EM1A\VALVE\CMD \$

SUBMODEL: EM1A\FLOW\CONTROLLER\DISCRETE ;

INPUTS: W2,W2SET ;

OUTPUTS: EM1A\VALVE\CMD \$

SUBMODEL: CORE\FLOW\CONTROLLER\CONTINUOUS ;

INPUTS: W13,W13SET ;

OUTPUTS: CORE\VALVE\CMD \$

SUBMODEL: CORE\FLOW\CONTROLLER\DISCRETE ;

INPUTS: W13,W13SET ;

OUTPUTS: CORE\VALVE\CMD \$

SUBMODEL: EM2B\FLOW\CONTROLLER\CONTINUOUS ;

INPUTS: W18,W18SET ;

OUTPUTS: EM2B\VALVE\CMD \$

SUBMODEL: EM2B\FLOW\CONTROLLER\DISCRETE ;

INPUTS: W18,W18SET ;

OUTPUTS: EM2B\VALVE\CMD \$

**** flow ****

* This file is located in /home/jacksme/marsyas/furn/flow on host ed14j. It

* is part of the main model and contains the flow and pressure dynamic

* equations used in the SSFF TCS simulation \$

* The following equations are written utilizing kirchoff's voltage law
 * and an equivalent electrical circuit analogy where pressure and flow are
 * analogous to voltage and current, respectively. The circuit drawing used
 * to obtain these equations is shown in the TECS Data Book (MSFC-DOC-2383)
 * in the controls analysis section. Certain paths (that is, w2, w3, w6, w13,
 * w17, and w18) have been modeled without flow inertance because a very large
 * eigenvalue results from the path's R/L combination. This causes the
 * simulation to be run at a very slow timestep. The flow inertances associated
 * with those paths are included in the upstream or downstream paths, hence
 * preserving the system inertance. An algebraic loop results in the no-
 * inertance paths between the calculation of flow and its resistance.
 * Algebraic loop breakers are added below to decrease simulation time. \$

* flow equations (lbm/hr) \$

EQUATION: $W1' = 1/L1*(PHP - P1 - PL1)$ \$

- : $W2D = (P3 - P4)/(RVF1A+R2)$ \$
- : $W2' = (W2D - W2)/(TDELAY)$ \$
- : $W3D = (P3 - P4)/(RMF1+R3)$ \$
- : $W3' = (W3D - W3)/(TDELAY)$ \$
- : $W4' = 1/L4*(P2 - P3 - PL4)$ \$
- : $W5' = 1/L5*(P4 - P5 - PL5)$ \$
- : $W6D = (P2 - P5)/(RMC+R6)$ \$
- : $W6' = (W6D - W6)/(TDELAY)$ \$
- : $W7' = 1/L7*(P1 - P2 - PL7)$ \$
- : $W8' = 1/L8*(P1 - P6 - PL8)$ \$
- : $W9' = 1/L9*(P5 - P9 - PL9)$ \$
- : $W10' = 1/L10*(P6 - P7 - PL10)$ \$
- : $W11' = 1/L11*(P7 - P8 - PL11)$ \$
- : $W12' = 1/L12*(P7 - P8 - PL12)$ \$
- : $W13D = (P8 - P9)/(R13+RVC)$ \$
- : $W13' = (W13D - W13)/(TDELAY)$ \$
- : $W14' = 1/L14*(P9 - P10 - PL14)$ \$
- : $W15' = 1/L15*(P6 - P11 - PL15)$ \$
- : $W16' = 1/L16*(P12 - P10 - PL16)$ \$
- : $W17D = (P11 - P12)/(R17 + RMF2)$ \$
- : $W17' = (W17D - W17)/(TDELAY)$ \$
- : $W18D = (P11 - P12)/(R18 + RVF2B)$ \$
- : $W18' = (W18D - W18)/(TDELAY)$ \$
- : $W19' = 1/L19*(P10 - PLP - PL19)$ \$

* pressure equations (psi) \$

- : $P1' = 1/C1*(W1 - W7 - W8)$ \$
- : $P2' = 1/C2*(W7 - W4 - W6)$ \$
- : $P3' = 1/C3*(W4 - W2 - W3)$ \$
- : $P4' = 1/C4*(W2 + W3 - W5)$ \$
- : $P5' = 1/C5*(W5 + W6 - W9)$ \$
- : $P6' = 1/C6*(W8 - W10 - W15)$ \$
- : $P7' = 1/C7*(W10 - W11 - W12)$ \$
- : $P8' = 1/C8*(W11 + W12 - W13)$ \$

```

: P9' = 1/C9*(W9 + W13 - W14) $
: P10' = 1/C10*(W14 + W16 - W19) $
: P11' = 1/C11*(W15 - W17 - W18) $
: P12' = 1/C12*(W17 + W18 - W16) $
: PLP' = 1/CACC*(W19 - W1) $

```

```

* pressure sensor equation (transfer function of 1) (psi) $
: PHEAD = P1 - P10 $

```

```

* equation for the high pressure side of the pump (psi) $
: PHP = PLP + PUMP(W1,PUMPASPEED,PUMP\REF\SPEED) $

```

```

* pressure losses in the flow paths (psi) $
: PL1 = W1*R1 $
: PL4 = W4*R4 $
: PL5 = W5*R5 $
: PL7 = W7*R7 $
: PL8 = W8*R8 $
: PL9 = W9*R9 $
: PL10 = W10*R10 $
: PL11 = W11*R11 $
: PL12 = W12*R12 $
: PL14 = W14*R14 $
: PL15 = W15*R15 $
: PL16 = W16*R16 $
: PL19 = W19*R19 $

```

```

* include the data for the flow and pressure equations $
INCLUDE: "data.flow" $

```

```

* convert time in seconds to time in minutes for output purposes$
: TIME\MIN = TIME/60 $
END $

```

```

**** defines ****

```

```

* This file is located in /home/jacksme/marsyas/furn/defines/ on host ed14j.
* It contains all the defined functions used in the SSFF TCS simulation $

```

```

* area of the pipe (ft^2) $
DEFINE: AREA (DIA) = PI * (DIA/2)**2 $

```

```

* calculation of the reynolds number (unitless). $
DEFINE: RE (FLOW,VISC) = (FABS(FLOW) / (3600*AREA(DC))) * DC / (VISC * GC) $

```

```

* table for viscosity of fluid (lbf-sec/ft^2) versus temperature (fah).
obtained from the textbook entitled "fluid mechanics" by streeter and
wylie (p711). $
DEFINE: VISC (TEMP) =
AF([ 0.0, 3.746E-5,

```

32.0, 3.746E-5,
 40.0, 3.229E-5,
 50.0, 2.735E-5,
 60.0, 2.359E-5,
 70.0, 2.050E-5,
 80.0, 1.799E-5,
 90.0, 1.595E-5,
 100.0, 1.424E-5,
 110.0, 1.284E-5,
 120.0, 1.168E-5,
 130.0, 1.069E-5,
 140.0, 0.981E-5,
 150.0, 0.905E-5], TEMP) \$

* empirical formula for calculating the friction factor where the relative line roughness (k/dn) equals 0.00008 (number supplied by boeing). formula obtained from a paper entitled "a theoretical manual for g.l. von pragenau's liquid seal simulation program" (may/june 1991) by dr. alan palazzolo. \$

DEFINE: FRICTION (RE) = 0.0055*(1+(20000.0*0.00008+(10**6)/(RE+1))**(1/3)) \$

* calculate kfactor given pressure drop @ a certain flow (unitless) \$

DEFINE: KFACTOR (PRESSURE, FLOW) = PRESSURE/FABS(FLOW)**2 * 2 * RHOF * (AREA(DC))**2 * GC * (3600**2) * 144 \$

* resistance (k-factor, flowrate) - calculate the resistance of a component (lbf-hr/lbm-in^2). \$

DEFINE: R (K, FLOW) = K*FABS(FLOW) / (2*RHOF*GC*144*(3600**2)*(AREA(DC))**2) \$

* table defining valve angle versus loss factor supplied by allied-signal \$

DEFINE: KVALVE (ANGLE) =
 AF([
 0.0,30000000.0,
 2.8, 5995463.0,
 5.0, 1199092.0,
 6.0, 599546.0,
 7.5, 257805.0,
 10.0, 101923.0,
 12.5, 38971.0,
 15.0, 21584.0,
 17.5, 11691.0,
 20.0, 8394.0,
 30.0, 1559.0,
 40.0, 539.6,
 50.0, 251.0,
 60.0, 125.0,
 75.0, 47.0,
 90.0, 22.0], ANGLE) \$

* table defining loss factor versus valve angle (degs) \$

```
DEFINE: ANGLE (K) =  
  AF([ 22.0, 90.0,  
      47.0, 75.0,  
      125.0, 60.0,  
      251.0, 50.0,  
      539.6, 40.0,  
      1559.0, 30.0,  
      8394.0, 20.0,  
      11691.0, 17.5,  
      21584.0, 15.0,  
      38971.0, 12.5,  
      101923.0, 10.0,  
      257805.0, 7.5,  
      599546.0, 6.0,  
      1199092.0, 5.0,  
      5995463.0, 2.8,  
      30000000.0, 0.0 ], K) $
```

* this is a pump map (lbm/hr versus psi) obtained at "rspeed" rpm.
rspeed/pspeed scales-up the flow to use a "rspeed" rmp map.
(pspeed/rspeed)**2 scales-down the output pressure to a "pspeed"
rmp map. \$

```
DEFINE: PUMP (FLOW,PSPEED,RSPEED) = ((PSPEED/RSPEED)**2) *  
  AF([ 0.0, 78.50,  
      250.2, 78.50,  
      500.5, 78.50,  
      1001.0, 78.50,  
      1251.2, 78.25,  
      1501.5, 78.00,  
      1751.7, 77.00,  
      2002.0, 76.00,  
      2252.2, 74.50,  
      2502.5, 72.50,  
      2752.7, 69.50,  
      3003.0, 66.00,  
      3253.2, 61.50,  
      3503.5, 54.00,  
      3603.6, 50.00 ], FABS(FLOW)*(RSPEED/PSPEED)) $
```

* k-factor for a fluid line \$

```
DEFINE: KLINE (FLOW,TEMP,LENGTH) = FRICTION((FABS(FLOW) /  
  (3600*AREA(DI))) * DI / (VISC(TEMP) * GC))*LENGTH/DI $
```

* resistance for a fluid line \$

```
DEFINE: RLINE (FLOW,TEMP,LENGTH) =  
  KLINE(FABS(FLOW),TEMP,LENGTH)*FABS(FLOW) /  
  (2*RHO*GC*144*(3600**2)*(AREA(DI)**2)) $
```

```

* convert watt to btu/sec $
DEFINE: QBTU(WATT) = WATT*3.4121/3600 $

* temperature sensor time constant (hr) versus flow (lbm/hr) $
*DEFINE: TC(FLOW) = 1/3600 * AF([ 29.96, 115.60,
    109.95, 68.03,
    230.44, 28.50,
    428.38, 12.94,
    812.40, 9.87,
    1224.14, 7.46], FLOW) $

```

```

* ua/ac vs mass flow for cp5 (btu/sec-f) $
*DEFINE: UA56(FLOW) =
    AC56/3600*AF([ 0.0, 725.0,
    25.0, 725.0,
    50.0, 830.0,
    100.0, 915.0,
    300.0, 1070.0,
    500.0, 1070.0], FLOW) $

```

***** data.flow *****

```

* This file is located in /home/jacksme/marsyas/furn/data.flow on host ed14j.
* It contains flow dynamic data and calculations for flow resistance
* and inertance for the SSFF TCS simulation $

```

```

* note: all pressures are in units of lbf/in^2 (psi)
    all flow rates are in units of lbm/hr $

```

```

* pump loss data $
EQUATION: P\PUMP = 16.7 $
    : MDPUMP = 3000 $
    : KPUMP = KFACTOR (P\PUMP, MDPUMP) $
* flow meter loss data $
    : PFMETER = 1.1 $
    : MDFMETER = 1300 $
    : KFMETER = KFACTOR (PFMETER, MDFMETER) $
* shut-off valve loss data $
    : PSOV = 0.2 $
    : MDSOV = 3000 $
    : KSOV = KFACTOR (PSOV, MDSOV) $
* temperature sensor loss data $
    : PTEMP = 0.29 $
    : MDTEMP = 600 $
    : KTEMP = KFACTOR (PTEMP, MDTEMP) $
* quick disconnect loss data $
    : PQD = 0.35 $
    : MDQD = 1230 $
    : KQD = KFACTOR (PQD, MDQD) $
* pressure transducer loss data $

```

: PPT = 1.29 \$
 : MDPT = 600 \$
 : KPT = KFACTOR (PPT, MDPT) \$
 * manual valve loss data \$
 * : PMAN = 0.25 \$
 * : MDMAN = 1230 \$
 * : KMAN = KFACTOR (PMAN, MDMAN) \$
 * water-to-water (ww) heat exchanger loss data \$
 : PWW = 0.5 \$
 : MDWW = 1100 \$
 : KWW = KFACTOR (PWW, MDWW) \$
 * water-to-air (wa) heat exchanger loss data \$
 : PWA = 1.0 \$
 : MDWA = 180 \$
 : KWA = KFACTOR (PWA, MDWA) \$
 * coldplate type cp-1-2-3 loss data \$
 : PCP123 = 0.6 \$
 : MDCP123 = 1000.0 \$
 : KCP123 = KFACTOR (PCP123, MDCP123) \$
 * coldplate type cp-4-5-6 loss data \$
 : PCP456 = 0.6 \$
 : MDCP456 = 300.0 \$
 : KCP456 = KFACTOR (PCP456, MDCP456) \$
 * coldplate type cp-7-8 loss data \$
 : PCP78 = 1.0 \$
 : MDCP78 = 300.0 \$
 : KCP78 = KFACTOR (PCP78, MDCP78) \$
 * furnace jacket loss data \$
 : PFURN = 0.7 \$
 : MDFURN = 150 \$
 : KFURN = KFACTOR (PFURN, MDFURN) \$
 * 3 foot flex line (1/2" dia) loss data \$
 : PFLEX = 0.77 \$
 : MDFLEX = 1230 \$
 : KFLEX = KFACTOR (PFLEX, MDFLEX) \$
 * 90 degree pipe bend loss data \$
 * : P90B = 0.038 \$
 * : MD90B = 3000 \$
 * : K90B = KFACTOR (P90B, MD90B) \$
 * resistance calculation for the flow control valves \$
 : RVF1A = R(KVALVE(EM1A\VALVE\ANGLE),W2) \$
 : RVC = R(KVALVE(CORE\VALVE\ANGLE),W13) \$
 : RVF2B = R(KVALVE(EM2B\VALVE\ANGLE),W18) \$
 * resistance calculation for the manual valves (static paths) \$
 : RMF1 = R(KVALVE(AMF1),W3) \$
 : RMC = R(KVALVE(AMC),W6) \$
 : RMF2 = R(KVALVE(AMF2),W17) \$
 * lumped pipe line lengths (ft) for each path \$
 : LL1 = 12.0/12 \$

: LL2 = 120.0/12 \$
 : LL3 = 72.0/12 \$
 : LL4 = 50.0/12 \$
 : LL5 = 52.0/12 \$
 : LL6 = 72.0/12 \$
 : LL7 = 20.0/12 \$
 : LL8 = 20.0/12 \$
 : LL9 = 40.0/12 \$
 : LL10 = 24.0/12 \$
 : LL11 = 24.0/12 \$
 : LL12 = 24.0/12 \$
 : LL13 = 24.0/12 \$
 : LL14 = 20.0/12 \$
 : LL15 = 26.5/12 \$
 : LL16 = 42.0/12 \$
 : LL17 = 72.0/12 \$
 : LL18 = 72.5/12 \$
 : LL19 = 146.0/12 \$

* lumped resistance calculations for each flow path (lbf-hr/lbm-in²) \$

: R1 = R(KPUMP,W1)/2 + R(KFLEX,W1) + R(KQD,W1) + R(KPT,W1)
 +RLINE(W1, T, LL1) \$
 : R2 = R(KFURN,W2) + 6*R(KQD,W2) + R(KFMETER,W2) + R(KTEMP,W2)
 +R(KSOV,W2)
 + 2*R(KFLEX,W2) + RLINE(W2, T, LL2) \$
 : R3 = R(KCP78,W3) + R(KCP456,W3) + 2*R(KCP78,W3) + R(KSOV,W3)
 + 12*R(KQD,W3) + 4*R(KFLEX,W3) + RLINE(W3, T, LL3) \$
 : R4 = RLINE(W4, T, LL4) \$
 : R5 = RLINE(W5, T, LL5) \$
 : R6 = 4*R(KCP456,W6) + R(KCP123,W6) + R(KCP78,W6) + 14*R(KQD,W6)
 + 2*R(KFLEX,W6) + RLINE(W6, T, LL6) \$
 : R7 = RLINE(W7, T, LL7) \$
 : R8 = RLINE(W8, T, LL8) \$
 : R9 = RLINE(W9, T, LL9) \$
 : R10 = R(KWA,W10) + 2*R(KQD,W10) + RLINE(W10, T, LL10) \$
 : R11 = R(KCP123,W11) + RLINE(W11, T, LL11) \$
 : R12 = R(KCP123,W12) + RLINE(W12, T, LL12) \$
 : R13 = R(KCP78,W13) + 2*R(KFLEX,W13) + R(KTEMP,W13)+4*R(KQD,W13)
 + R(KFMETER,W13) + RLINE(W13, T, LL13) \$
 : R14 = RLINE(W14, T, LL14) \$
 : R15 = RLINE(W15, T, LL15) \$
 : R16 = RLINE(W16, T, LL16) \$
 : R17 = R(KCP78,W17) + R(KCP456,W17) + 2*R(KCP78,W17) + R(KSOV,W17)
 + 12*R(KQD,W17) + 4*R(KFLEX,W17) + RLINE(W17, T, LL17) \$
 : R18 = R(KFURN,W18) + 6*R(KQD,W18) + R(KFMETER,W18)
 +R(KTEMP,W18)
 + R(KSOV,W18) + 2*R(KFLEX,W18) + RLINE(W18, T, LL18) \$
 : R19 = R(KPUMP,W19)/2 + R(KWW,W19) + R(KPUMP,W19)/2 + R(KQD,W19)
 + 3*R(KFLEX,W19) + R(KPT,W19) + RLINE(W19, T, LL19) \$

* line cross sectional area (ft²) \$

: A = AREA(DI) \$

* flex line length (ft) \$

: FLEX = 3.0 \$

* lumped line inertances (hr-sec-lbf/lbm-in²) plus equivalent rack inertances
calculations for each flow path \$

: L1 = (LL1+FLEX)/(3600*144*GC*A) \$

: L4 = LL4/(3600*144*GC*A) + LEQ1/2 \$

: L5 = LL5/(3600*144*GC*A) + LEQ1/2 \$

: L7 = LL7/(3600*144*GC*A) + L6/2 \$

: L8 = LL8/(3600*144*GC*A) \$

: L9 = LL9/(3600*144*GC*A) + L6/2 \$

: L10 = LL10/(3600*144*GC*A) \$

: L11 = LL11/(3600*144*GC*A) + L13/2 \$

: L12 = LL12/(3600*144*GC*A) + L13/2 \$

: L14 = LL14/(3600*144*GC*A) \$

: L15 = LL15/(3600*144*GC*A) + LEQ2/2 \$

: L16 = LL16/(3600*144*GC*A) + LEQ2/2 \$

: L19 = (LL19+2*FLEX)/(3600*144*GC*A) \$

: L2 = (LL2+2*FLEX)/(3600*144*GC*A) \$

: L3 = (LL3+4*FLEX)/(3600*144*GC*A) \$

: L6 = (LL6+2*FLEX)/(3600*144*GC*A) \$

: L13 = (LL13+2*FLEX)/(3600*144*GC*A) \$

: L17 = (LL17+2*FLEX)/(3600*144*GC*A) \$

: L18 = (LL18+2*FLEX)/(3600*144*GC*A) \$

: LEQ1 = L2*L3/(L2+L3) \$

: LEQ2 = L17*L18/(L17+L18) \$

* node capacitances (lbf-hr/lbm-in²-sec) \$

: C1 = 0.1 \$

: C2 = 0.1 \$

: C3 = 0.1 \$

: C4 = 0.1 \$

: C5 = 0.1 \$

: C6 = 0.1 \$

: C7 = 0.1 \$

: C8 = 0.1 \$

: C9 = 0.1 \$

: C10 = 0.1 \$

: C11 = 0.1 \$

: C12 = 0.1 \$

: CACC = 0.1 \$

**** f_act ****

* This file is located in /home/jacksme/marsyas/furn/f_act on host ed14j.

* It contains the actuator/valve models for each flow control path \$

MODEL: EM1A\VALVEACTUATOR \$

```

INPUTS: VA $
OUTPUTS: AV $
* wv - valve speed response to the input speed voltage command (va) $
* tau - delay between commanded and actual valve speed response (secs) $
* kv - voltage to speed conversion constant (degs/sec/volts) $
* av - valve position (degs) $
EQUATION: WV' = 1/TAU*(KV*VA - WV) $
: AV' = WV $
: KV = 1.0 $
: TAU = 0.01 $
END $

```

```

MODEL: CORE\VALVE\ACTUATOR $
INPUTS: VA $
OUTPUTS: AV $
* wv - valve speed response to the input speed voltage command (va) $
* tau - delay between commanded and actual valve speed response (secs) $
* kv - voltage to speed conversion constant (degs/sec/volts) $
* av - valve position (degs) $
EQUATION: WV' = 1/TAU*(KV*VA - WV) $
: AV' = WV $
: KV = 1.0 $
: TAU = 0.01 $
END $

```

```

MODEL: EM2B\VALVE\ACTUATOR $
INPUTS: VA $
OUTPUTS: AV $
* wv - valve speed response to the input speed voltage command (va) $
* tau - delay between commanded and actual valve speed response (secs) $
* kv - voltage to speed conversion constant (degs/sec/volts) $
* av - valve position (degs) $
EQUATION: WV' = 1/TAU*(KV*VA - WV) $
: AV' = WV $
: KV = 1.0 $
: TAU = 0.01 $
END $

```

**** f_cont ****

```

* This file is located in /home/jacksme/marsyas/furn/f_cont on host ed14j.
* This file contains the submodels for the continuous and discrete
* flow control algorithms. First, the controllers were designed in the
* discrete domain and then converted to the continuous domain using
* pole-zero mapping (ref. Discrete Time Control Systems by Ogata). $

```

```

* em1a discrete flow control submodel $
MODEL: EM1A\FLOW\CONTROLLER\DISCRETE,DISCRETE $
INPUTS: W2,W2SET $
OUTPUTS: EM1A\VALVE\CMD $

```

* bumpless transfer function - digital low pass filter that does not allow the set point to change rapidly \$

EQUATION: $BSET = 0.90*BSETOLD + 0.10*W2SET$ \$
: $BSETOLD\% = BSET$ \$

* flow error based on filtered set-point \$
: $ERR = BSET - W2$ \$

* digital low pass noise filter does not allow the current error to effect the computed output error by more than 10 percent. the previously computed output error comprises 90 percent of the computed output error \$

: $FERR = 0.90*FERROLD + 0.10*ERR$ \$
: $FERROLD\% = FERR$ \$

* compute error derivative \$
: $FEDOT = (FERR - FERROLD)/1.0$ \$

* velocity pd control algorithm \$
: $U = FERR*0.015 + FEDOT*0.1$ \$

* valve command limiter \$
: $VCMD = 5.0$ IF U.GT.5.0 ELSE
-5.0 IF U.LT.-5.0 ELSE
U \$
: $EM1A\VALVE\CMD = VCMD$ \$

END \$

* em1a continuous flow control submodel \$

MODEL: EM1A\FLOWCONTROLLER\CONTINUOUS \$

INPUTS: W2,W2SET \$

OUTPUTS: EM1A\VALVE\CMD \$

* bumpless transfer function - digital low pass filter that does not allow the set point to change rapidly \$

EQUATION: $BW2SET' = -.1054*BW2SET + .1054*W2SET$ \$

* flow error based on filtered set-point \$
: $ERR = BW2SET - W2$ \$

* digital low pass noise filter does not allow the current error to effect the computed output error by more than 10 percent. the previously computed output error comprises 90 percent of the computed output error \$

: $FERR = TF(1,0,0.1054,1.0,0.1054,ERR)$ \$

* velocity pd control algorithm \$
: $U1A = TF(1,1,.015,TDELAY,1.0,FERR)$ \$

* valve command limiter \$
: $VCMD1A = 5.0$ IF U1A.GT.5.0 ELSE
-5.0 IF U1A.LT.-5.0 ELSE
U1A \$

* computer delay (= 1.0 sec) \$
: $EM1A\VALVE\CMD' = -2*EM1A\VALVE\CMD + 2*VCMD1A$ \$

END \$

* core discrete flow control submodel \$

MODEL: CORE\FLOWCONTROLLER\DISCRETE,DISCRETE \$

INPUTS: W13,W13SET \$

OUTPUTS: CORE\VALVE\CMD \$

* bumpless transfer function - digital low pass filter that does not allow the set point to change rapidly \$

EQUATION: BSET = 0.90*BSETOLD + 0.10*W13SET \$
: BSETOLD% = BSET \$

* flow error based on filtered set-point \$
: ERR = BSET - W13 \$

* digital low pass noise filter does not allow the current error to effect the computed output error by more than 10 percent. the previously computed output error comprises 90 percent of the computed output error \$

: FERR = 0.90*FERROLD + 0.10*ERR \$
: FERROLD% = FERR \$

* compute error derivative \$
: FEDOT = (FERR - FERROLD)/1.0 \$

* velocity pd control algorithm \$
: U = FERR*0.015 + FEDOT*0.1 \$

* valve command limiter \$
: VCMD = 5.0 IF U.GT.5.0 ELSE
-5.0 IF U.LT.-5.0 ELSE
U \$

: CORE\VALVE\CMD = VCMD \$

END \$

* core continuous flow control submodel \$

MODEL: CORE\FLOW\CONTROLLER\CONTINUOUS \$

INPUTS: W13,W13SET \$

OUTPUTS: CORE\VALVE\CMD \$

* bumpless transfer function - digital low pass filter that does not allow the set point to change rapidly \$

EQUATION: BW13SET' = -.1054*BW13SET + .1054*W13SET \$

* flow error based on filtered set-point \$
: ERR = BW13SET - W13 \$

* digital low pass noise filter does not allow the current error to effect the computed output error by more than 10 percent. the previously computed output error comprises 90 percent of the computed output error \$

: FERR = TF(1,0,0.1054,1,0,0.1054,ERR) \$

* velocity pd control algorithm \$
: U = TF(1,.,1,.,015,TDELAY,1,0,FERR) \$

* valve command limiter \$
: VCMD = 5.0 IF U.GT.5.0 ELSE
-5.0 IF U.LT.-5.0 ELSE
U \$

* computer delay (= 1.0 sec) \$
: CORE\VALVE\CMD' = -2*CORE\VALVE\CMD + 2*VCMD \$

END \$

* em2b discrete flow control submodel \$

MODEL: EM2B\FLOW\CONTROLLER\DISCRETE,DISCRETE \$

```

INPUTS: W18,W18SET $
OUTPUTS: EM2B\VALVE\CMD $
* bumpless transfer function - digital low pass filter that does not
  allow the set point to change rapidly $
EQUATION: BSET = 0.90*BSETOLD + 0.10*W18SET $
          : BSETOLD% = BSET $
* flow error based on filtered set-point $
          : ERR = BSET - W18 $
* digital low pass noise filter does not allow the current error
  to effect the computed output error by more than 10 percent.
  the previously computed output error comprises 90 percent of
  the computed output error $
          : FERR = 0.90*FERROLD + 0.10*ERR $
          : FERROLD% = FERR $
* compute error derivative $
          : FEDOT = (FERR - FERROLD) / 1.0 $
* velocity pd control algorithm $
          : U = FERR*0.015 + FEDOT*0.1 $
* valve command limiter $
          : VCMD = 5.0 IF U.GT.5.0 ELSE
            -5.0 IF U.LT.-5.0 ELSE
              U $
          : EM2B\VALVE\CMD = VCMD $
END $

```

```

* em2b continuous flow control submodel $
MODEL: EM2B\FLOW\CONTROLLER\CONTINUOUS $
INPUTS: W18,W18SET $
OUTPUTS: EM2B\VALVE\CMD $
* bumpless transfer function - digital low pass filter that does not
  allow the set point to change rapidly $
EQUATION: BW18SET' = -.1054*BW18SET + .1054*W18SET $
* flow error based on filtered set-point $
          : ERR = BW18SET - W18 $
* digital low pass noise filter does not allow the current error
  to effect the computed output error by more than 10 percent.
  the previously computed output error comprises 90 percent of
  the computed output error $
          : FERR = TF(1,0,0.1054,1.0,0.1054,ERR) $
* velocity pd control algorithm $
          : U = TF(1,.,.015,TDELAY,1.0,FERR) $
* valve command limiter $
          : VCMD = 5.0 IF U.GT.5.0 ELSE
            -5.0 IF U.LT.-5.0 ELSE
              U $
* computer delay (= 1.0 sec) $
          : EM2B\VALVE\CMD' = -2*EM2B\VALVE\CMD + 2*VCMD $
END $

```

```

**** p_cont ****

```

* This file is located in /home/jacksme/marsyas/furn/p_cont on host ed14j.

* It contains the continuous and discrete pressure control algorithms \$

* continuous pressure controller \$

MODEL: PRESSURE\CONTROLLER\CONTINUOUS \$

INPUTS: PHEAD,PHEADSET \$

OUTPUTS: PUMP\SPEED\CMD \$

EQUATION : PERR = PHEADSET - PHEAD \$

* controller transfer function =

* $ko*(s^2 + (br2+br3)*s + br2*br3) / (s^2 + (br1+br4)*s + br1*br4)$ \$

EQUATION : $X1' = -A1*X1 + X2 + (KO*B1 - A1*KO)*PERR$ \$

: $X2' = KO*BO*PERR$ \$

: $CMD = (X1 + KO*PERR)*2*PI/60$ \$

* controller coefficient definitions \$

: $A1 = BR1+BR4$ \$

: $B1 = BR2+BR3$ \$

: $BO = BR2*BR3$ \$

* controller gain (volts/psi) \$

: $KO = 6000$ \$

* controller break frequencies (rads/sec) \$

: $BR1 = 0.0$ \$

: $BR2 = 0.1*2*PI$ \$

: $BR3 = 1.0*2*PI$ \$

: $BR4 = 10.0*2*PI$ \$

* computer delay (= 1.0 sec) \$

: $PUMP\SPEED\CMD' = -2*PUMP\SPEED\CMD + 2*CMD$ \$

END \$

* discrete pressure controller \$

MODEL: PRESSURE\CONTROLLER\DISCRETE,DISCRETE \$

INPUTS: PHEAD,PHEADSET \$

OUTPUTS: PUMP\SPEED\CMD \$

EQUATION: $E = PHEADSET - PHEAD$ \$

* difference equation obtained by converting the continuous pressure controller to the discrete domain using pole-zero mapping (ref. Discrete Time Control Systems by Ogata) \$

: $U = UD + KSTAR*(E - (EXP(-BR2*T)+EXP(-BR3*T))*ED\% + EXP(-BR2*T)*EXP(-BR3*T)*ED)$ \$

: $ED\%\% = E$ \$

: $UD\% = U$ \$

* discrete controller gain (volts/psi) \$

: $KSTAR = KO*(BR2+BR3)/(BR4*(2-EXP(-BR2*T)-EXP(-BR3*T)))$ \$

* continuous controller gain (volts/psi) \$

: $KO = 6000$ \$

* continuous controller break frequencies (rads/sec) \$

: $BR2 = 0.1*2*PI$ \$

: $BR3 = 1.0*2*PI$ \$

: $BR4 = 10.0*2*PI$ \$

: $T = 1.0$ \$

```
* convert pump speed from rev/min to rad/sec $
: PUMP\SPEED\CMD = U*2*PI/60 $
END $
```

```
**** pump_sc ****
```

```
* This file is located in /home/jacksme/marsyas/furn/pump_sc on host ed14j.
* It contains the pump speed controller model supplied by Allied Signal
* (pump vendor). $
```

```
MODEL: PUMP\SPEED\CONTROLLER $
INPUTS: PUMP\SPEED\CMD,PUMP\SPEED\SEN $
OUTPUTS: PUMP\VOLT\CMD $
* speed command limiter in (radians/second) $
EQUATION: SPEED\CMD\LIM = 22222.0*(2*PI/60) IF PUMP\SPEED\CMD.GT.22222.0*(2*PI/60)
ELSE
    0.0 IF PUMP\SPEED\CMD.LT.0.0 ELSE
    PUMP\SPEED\CMD $
* sensed speed filter $
: SPEED\FBK' = (PUMP\SPEED\SEN - SPEED\FBK)/0.00796 $
* calculate speed error (radians/second) $
: SPEED\ERR = (SPEED\CMD\LIM - SPEED\FBK)/2327.1 $
* speed controller $
* : VOLT = 120*TF(1,.124,1,.3,0,SPEED\ERR) $
: X1' = SPEED\ERR $
: VOLT = 120.0/.3*(.124*SPEED\ERR + X1) $
* voltage limiter $
: PUMP\VOLT\CMD = 120 IF VOLT.GT.120 ELSE
    0 IF VOLT.LT.0 ELSE
    VOLT $
END $
```

```
**** pump_motor ****
```

```
* This file is located in /home/jacksme/marsyas/furn/pump_motor on host ed14j.
* It contains the pump motor and speed sensor model supplied by Applied
* Signal (pump vendor). $
```

```
MODEL: PUMP\MOTOR $
INPUTS: PUMP\VOLT\CMD $
OUTPUTS: PUMP\SPEED,PUMP\SPEED\SEN,PUMP\CURR\SEN $
EQUATION: VOLTERR = PUMP\VOLT\CMD - VEMF $
* back emf gain (volt/rads/sec) $
: KE = 0.04185 $
* back emf calculation (volts) $
: VEMF = KE*PUMP\SPEED\SEN $
* lm - motor inductance (henries) $
: LM = 0.0012 $
* rm - motor resistance (ohms) $
: RM = 0.618 $
```

```

* motor impedance transfer function $
*   : PUMP\CURR\SEN = TF(1,0.0,1/RM,LM/RM,1.0,VOLTERR) $
*   : PUMP\CURR\SEN' = -RM/LM*PUMP\CURR\SEN + 1/LM*VOLTERR $
* torque gain (in-lb/amp) $
*   : KT = 0.33125 $
* motor torque calculation (in-lb) $
*   : TM = KT*PUMP\CURR\SEN $
* actual torque calculation (in-lb) $
*   : TA = TM - TL $
* empirical torque load calculation (in-lb) $
*   : TL = 1.9875*(PUMP\SPEED\SEN)**2/1979**2 $
* rotor and load inertia (in-lb-s^2) $
*   : JML = 500E-6/16 $
* derivative of angular speed equals torque/inertia (rads/sec) $
*   : PUMP\SPEED\SEN' = TA/JML $
* output equation to pump map $
*   : PUMP\SPEED = PUMP\SPEED\SEN*60/(2*PI) $
END $

```

**** sim ****

```

* This file is located in /home/jacksme/marsyas/furn/sim on host ed14j.
* This file contains the simulation module for the main SSFF TCS model.
* It includes the input excite functions, the initialization statements,
* the discrete submodel schedules, etc. $

```

SIMULATE: MAIN \$

```

* excitation of the inputs in the main model $

```

```

EXCITE: AMF1 = 2.130711E+01 $
*   : AMC = 2.123931E+01 $
*   : AMF2 = 2.123383E+01 $
*   : PUMP\REFSPEED = 18900.0 $
*   : T = 80.0 $
*   : PHEADSET = 10.0 $
*   : W2SET = 260.0 $
*   : W13SET = 386.0 $
*   : W18SET = 260.0 $

```

```

* set initial values for the states in the main model $

```

```

INITIALIZE: MAIN, PLP = 2.600000E+01,
*   W1 = 1.056000E+03,
*   W2 = 2.600000E+02,
*   W3 = 5.000000E+01,
*   W4 = 3.100000E+02,
*   W5 = 3.100000E+02,
*   W6 = 5.000000E+01,
*   W7 = 3.600000E+02,
*   W8 = 6.960000E+02,
*   W9 = 3.600000E+02,
*   W10 = 3.860000E+02,

```

W11 = 193.0,
W12 = 193.0,
W13 = 3.860000E+02,
W14 = 7.460000E+02,
W15 = 3.100000E+02,
W16 = 3.100000E+02,
W17 = 5.000000E+01,
W18 = 2.600000E+02,
W19 = 1.056000E+03,

P1 = 4.513639E+01,
P2 = 4.512252E+01,
P3 = 4.509572E+01,
P4 = 3.524069E+01,
P5 = 3.521281E+01,
P6 = 4.509319E+01,
P7 = 4.040686E+01,
P8 = 4.037880E+01,
P9 = 3.518509E+01,
P10 = 3.513639E+01,
P11 = 4.507899E+01,
P12 = 3.515890E+01 \$

* set initial values in the pump speed controller submodel \$

INITIALIZE: PUMP\SPEED\CONTROLLER, SPEED\FBK = 18900.0*5.717402E-01*(2*PI/60) \$
: PUMP\SPEED\CONTROLLER, XI = 1.214235E-01 \$

* set initial values in the pump motor submodel \$

INITIALIZE: PUMP\MOTOR, PUMP\CURRISEN = 1.961725E+00 \$
: PUMP\MOTOR, PUMP\SPEED\SEN = 18900.0*5.717402E-01*(2*PI/60) \$

* set initial values in the valve actuator submodel \$

INITIALIZE: EM1A\VALVE\ACTUATOR, WV = 0.0, AV = 5.310246E+01 \$
: CORE\VALVE\ACTUATOR, WV = 0.0, AV = 7.984484E+01 \$
: EM2B\VALVE\ACTUATOR, WV = 0.0, AV = 5.291287E+01 \$

* set initial values in the continuous pressure controller submodel \$

INITIALIZE: PRESSURE\CONTROLLER\CONTINUOUS,
X1=18900.0*5.717402E-01,
X2=18900.0*5.717402E-01*10*2*PI,
PUMP\SPEED\CMD = (18900.0*5.717402E-01)*2*PI/60 \$

* set initial values in the discrete pressure controller submodel \$

INITIALIZE: PRESSURE\CONTROLLER\DISCRETE, UD = 18900.0*5.717402E-01 \$

* set initial value in the continuous and discrete flow controllers \$

INITIALIZE: EM1A\FLOW\CONTROLLER\CONTINUOUS, BW2SET = 260.0 \$
INITIALIZE: CORE\FLOW\CONTROLLER\CONTINUOUS, BW13SET = 386.0 \$
INITIALIZE: EM2B\FLOW\CONTROLLER\CONTINUOUS, BW18SET = 260.0 \$
INITIALIZE: EM1A\FLOW\CONTROLLER\DISCRETE, BSETOLD = 260.0 \$

INITIALIZE: CORE\FLOW\CONTROLLER\DISCRETE, BSETOLD = 386.0 \$
INITIALIZE: EM2B\FLOW\CONTROLLER\DISCRETE, BSETOLD = 260.0 \$

* numerically integrate using runga-kutta 4th order method \$
INTEGRATE: RK4, TIMESTEP, TSTEP \$

* set simulation stop time \$
STOPIF : TIME.GE.STOPT \$

* plot the main outputs every 100 integration steps \$
PLOT : LINEAR(0,STOPT,100),\OUTPUTS\ \$

* schedule each discrete controller every 1.0 second \$
SCHEDULE: PRESSURE\CONTROLLER\DISCRETE,FMOD(TIME+TSTEP/2,1.0).LT.TSTEP \$
SCHEDULE: EM1A\FLOW\CONTROLLER\DISCRETE,FMOD(TIME+TSTEP/2,1.0).LT.TSTEP \$
SCHEDULE: CORE\FLOW\CONTROLLER\DISCRETE,FMOD(TIME+TSTEP/2,1.0).LT.TSTEP \$
SCHEDULE: EM2B\FLOW\CONTROLLER\DISCRETE,FMOD(TIME+TSTEP/2,1.0).LT.TSTEP \$

END \$

APPROVAL

**PRELIMINARY CONTROL SYSTEM DESIGN AND ANALYSIS FOR THE
SPACE STATION FURNACE FACILITY
THERMAL CONTROL SYSTEM**

By M.E. Jackson

The information in this report has been reviewed for technical content. Review of any information concerning Department of Defense or nuclear energy activities or programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.



J.C. BLAIR
Director, Structures and Dynamics Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE January 1995	3. REPORT TYPE AND DATES COVERED Technical Memorandum	
4. TITLE AND SUBTITLE Preliminary Control System Design and Analysis for the Space Station Furnace Facility Thermal Control System			5. FUNDING NUMBERS
6. AUTHOR(S) M.E. Jackson			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) George C. Marshall Space Flight Center Marshall Space Flight Center, Alabama 35812			8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546			10. SPONSORING / MONITORING AGENCY REPORT NUMBER NASA TM-108476
11. SUPPLEMENTARY NOTES Prepared by Structures and Dynamics Laboratory, Science and Engineering Directorate.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Unclassified—Unlimited			12b. DISTRIBUTION CODE
13. ABSTRACT (Maximum 200 words) This report presents the Space Station Furnace Facility (SSFF) thermal control system (TCS) preliminary control system design and analysis. The SSFF provides the necessary core systems to operate various materials processing furnaces. The TCS is defined as one of the core systems, and its function is to collect excess heat from furnaces and to provide precise cold temperature control of components and of certain furnace zones. Physical interconnection of parallel thermal control subsystems through a common pump implies the description of the TCS by coupled nonlinear differential equations in pressure and flow. This report formulates the system equations and develops the controllers that cause the interconnected subsystems to satisfy flow rate tracking requirements. Extensive digital simulation results are presented to show the flow rate tracking performance.			
14. SUBJECT TERMS Thermal Control Systems, Nonlinear Systems, Digital Control, Frequency Response			15. NUMBER OF PAGES 38
			16. PRICE CODE NTIS
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT Unlimited

National Aeronautics and
Space Administration
Code JTT
Washington, DC
20546-0001

*Official Business
Penalty for Private Use, \$300*

Postmaster: If Undeliverable (Section 158 Postal Manual), Do Not Return
